



Національний університет
водного господарства
та природокористування

Міністерство освіти і науки України
Національний університет водного господарства
та природокористування
Кафедра обчислювальної техніки

04-04-224

МЕТОДИЧНІ ВКАЗІВКИ ТА ЗАВДАННЯ

до виконання лабораторних робіт з навчальної дисципліни
«Програмування засобами C#.NET» для здобувачів вищої освіти другого
(магістерського) рівня за спеціальністю 123 "Комп'ютерна інженерія"
денної та заочної форм навчання

Рекомендовано
науково-методичною комісією
зі спеціальності 123 "Комп'ютерна інженерія".
Протокол № 1 від 10.09.2018 р.

Рівне , 2018

Методичні вказівки та завдання до виконання лабораторних робіт з навчальної дисципліни «Програмування засобами C#.NET» для здобувачів вищої освіти другого (магістерського) рівня за спеціальністю 123 "Комп'ютерна інженерія" денної та заочної форм навчання /Ольшанський П.В. : НУВГП, 2018. – 32 с.

Укладач: Ольшанський П.В. – старший викладач кафедри
обчислювальної техніки.

Відповідальний за випуск – Б.Б. Круліковський , канд. техн. наук,
доцент, завідувач кафедри
обчислювальної техніки.



Національний університет
водного господарства
та природокористування

© Ольшанський П.В., 2018
© Національний університет
водного господарства та
природокористування, 2018



Тема. Розробка простого консольного застосунку.

Мета. Ознайомлення з середовищем Microsoft Visual Studio; ознайомлення зі структурою консольного застосунку; вивчення базової структури програми на C#.

Завдання

Розробити консольний застосунок, який розраховуватиме заданий математичний вираз з однією змінною $f(x)=x^2$ в певному діапазоні. Програма повинна забезпечувати ввід мінімального x_{min} та максимального x_{max} значення x , а також кроку його зміни dx . Всі розраховані значення виразу для всіх $x_{min} \leq x \leq x_{max}$ слід послідовно вивести на екран.

Виконання завдання

1. В середовищі Microsoft Visual Studio створити новий проект консольного застосунку. Для цього вибрати команду меню File/New/Project. У вікні New Project (рис. 1.1) вибрати шаблон ConsoleApplication у розділі Templates/ Visual C#/ Windows. Далі вказати назву проекту (наприклад, Lab01, але можна задати іншу) та його розташування на диску.

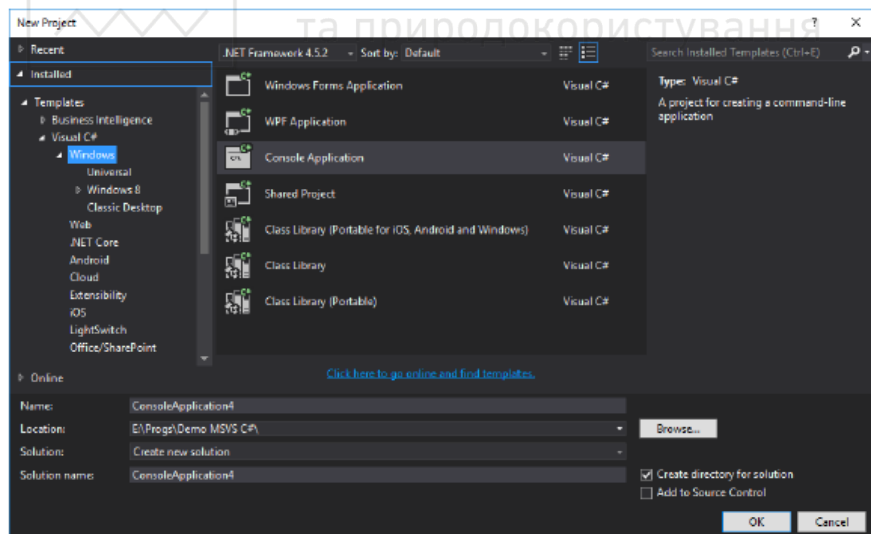


Рис. 1.1. Вікно для задання параметрів нового проекту

2. У вікні Solution Explorer ознайомитися зі структурою проекту (рис. 1.2). Зайти у папку, де було збережено проект, і ознайомитися з його файловою структурою. В загальному випадку проект типу Console Application містить такі частини:

- Папка Lab01 – папка для рішення (solution)
 - a. Файл Lab01.sln – xml-файл для опису параметрів рішення (файл рішення)
 - b. Папка Lab01 – папка проекту застосунку
 - i. Файл Lab01.csproj – xml-файл для опису параметрів проекту (файл проекту)
 - ii. Папка Properties – для зберігання властивостей проекту. Розглядати файл AssemblyInfo.cs зараз не будемо. У ньому міститься додаткова інформація про проект
 - iii. Файл Program.cs – код головного модуля застосунку мовою C#

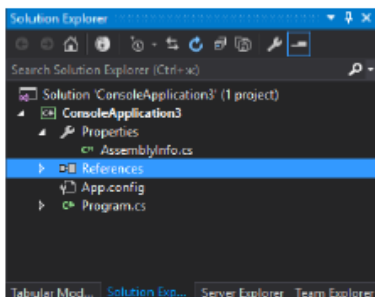


Рис. 1.2. Вікно Solution Explorer

3. Продумати алгоритм програми. Алгоритм для виконання завдання такий:
- 1) Ввести $xMin$, $xMax$, dx .
 - 2) Тимчасовій змінній x присвоїти значення $xMin$.
 - 3) У циклі, доки $x \leq xMax$:
 - Розрахувати значення функції $y = f(x)$;
 - Вивести на екран пару: x та y ;
 - Збільшити значення x на dx ;
 - 4) Якщо $x < xMax$, то:
 - Розрахувати значення функції $y = f(xMax)$;
 - Вивести на екран пару: $xMax$ та y ;
4. У редакторі коду Microsoft Visual Studio у функцію Main() записати код з лістингу 1.1.
5. Запустити застосунок (F5 в середовищі Microsoft Visual Studio) і перевірити його роботу. Приклад виконання розробленої програми приведено на рис. 1.3.

Лістинг 1.1

```
Console.WriteLine("Введіть початкове значення Xmin: ");
string sxMin = Console.ReadLine();
double xMin = Double.Parse(sxMin);

Console.WriteLine("Введіть кінцеве значення Xmax: ");
string sxMax = Console.ReadLine();
double xMax = double.Parse(sxMax);

Console.WriteLine("Введіть приріст dx: ");
string sdx = Console.ReadLine();
double dx = double.Parse(sdx);

double x = xMin;
double y;

while (x <= xMax)
{
    y = Math.Pow(x, 2);
    Console.WriteLine("x = {0}\t\t y = {1}", x, y);

    x += dx;
```

```

}
}
if (Math.Abs(xMax - dx) > 0.0001)
{
    y = Math.Pow(xMax, 2);
    Console.WriteLine("x = {0}\t\t y = {1}", xMax, y);
}

Console.ReadKey();

```

```

C:\Users\Bart\Documents\Visual Studio 2010\Projects\Lab01\Debug\Lab01.P38
Введіть початкове значення Xmin: 3,892
Введіть кінцеве значення Xmax: 142,984
Введіть приріст dx: 6,271
x = 3,892          y = 15,147664
x = 10,163         y = 103,286569
x = 16,434         y = 270,076356
x = 22,705         y = 515,517025
x = 28,976         y = 839,688576
x = 35,247         y = 1242,351009
x = 41,518         y = 1723,744324
x = 47,789         y = 2283,788521
x = 54,06          y = 2922,4836
x = 60,331         y = 3639,829561
x = 66,602         y = 4435,826404
x = 72,873         y = 5310,474129
x = 79,144         y = 6263,772736
x = 85,415         y = 7295,722225
x = 91,686         y = 8406,322596
x = 97,957         y = 9595,573849
x = 104,228        y = 10863,475984
x = 110,499        y = 12210,029001
x = 116,77         y = 13635,2329
x = 123,041        y = 15139,087681
x = 129,312        y = 16721,593344
x = 135,583        y = 18382,748889

```

Рис. 1.3. Вивід розробленого застосунку у консоль

Завдання для самостійного опрацювання

- Вирішити розглянуте раніше завдання для заданої функції $f(x)$ згідно варіанту (таблиця 1.1). У виразах прийняти, що $x_1=x$, $x_2=3x$.
- Додатково до попереднього завдання, обчислити та вивести на екран:
 - для варіантів, номери яких закінчуються на 0 чи 5 – суму всіх розрахованих проміжних значень $f(x)$;
 - для варіантів, номери яких закінчуються на 1 чи 6 – добуток всіх розрахованих проміжних значень $f(x)$;
 - для варіантів, номери яких закінчуються на 2 чи 7 – суму синусів всіх розрахованих проміжних значень $f(x)$;
 - для варіантів, номери яких закінчуються на 3 чи 8 – добуток косинусів всіх розрахованих проміжних значень $f(x)$;
 - для варіантів, номери яких закінчуються на 4 чи 9 – суму кубів всіх розрахованих проміжних значень $f(x)$;



Варіант	Завдання	Варіант	Завдання
1.	$y = \frac{\sqrt{x_1^3 + x_2^5} + 65}{1000}$	2.	$y = \sqrt{56x_1 + \frac{x_1 + x_2 + \sin(x_1x_2)}{5 - \cos(x_2^2)}}$
3.	$y = \cos(x_1 - x_2^2) + 31.55x_2x_1^2$	4.	$y = \sqrt[3]{0.1x_1 \sin x_2 \cos x_1^2 + 55x_1x_2}$
5.	$y = \frac{6 - \cos(3 + x_1)}{34 - 9x_2^3 + x_2}$	6.	$y = \sqrt{\frac{\cos(2x_2) + x_1/x_2}{16x_2x_1}}$
7.	$y = \cos^4\left(x_1 - \sqrt{\frac{x_2}{x_1 + 53x_2^2}}\right)$	8.	$y = \cos(\sqrt{x_2 + 34x_1}) - 4\sin(x_2)$
9.	$y = 23\sin^2(x_1^3x_2^5) + 2x_1 + \cos(x_1x_2)$	10.	$y = \sin^2\left(x_1 \frac{x_2}{x_1 + 53x_2^2}\right)$
11.	$y = \sqrt{56 + \frac{x_1 + x_2 + \sin(x_1x_2)}{5\cos(x_2^2)}}$	12.	$y = \frac{3x_2 - x_1^2}{\cos^3\left(\exp\left(\frac{x_1 + 2x_2 + 9}{0.37}\right)\right)}$
13.	$y = 45x_1 \sin x_2 + \sqrt{9x_2x_1^3}$	14.	$y = \cos(\sqrt{x_2 + 34 \cdot \sin(x_1)}) - 4\sin(x_2)$
15.	$y = \frac{1}{4 + x_2} \cdot \sqrt{\cos^2\left(\exp \frac{x_2}{x_1}\right)}$	16.	$y = \sqrt{\frac{x_2^2 + x_1/x_2}{\cos(x_1^3x_2^3) + 2x_1}}$
17.	$y = \frac{\cos^2(x_1) + 45 + x_2}{x_1^{13} + \cos(x_2)}$	18.	$y = 0.1x_1 \sin x_2 \cos x_1^4 + 55$
19.	$y = \frac{5\sqrt{x_1^3 + x_2^5} - \cos(x_2)}{\sin(x_1)}$	20.	$y = \frac{\sin^3(x_1) + 45 + x_2}{2x_1^4 + 4x_2}$
21.	$y = \frac{\sqrt{x_1^3 + x_2^5}}{1000 \cdot \sqrt{x_1 + x_2^5}} + 65$	22.	$y = \frac{\cos^2\left(\lg_{10} \frac{x_2}{x_1}\right)}{45 + x_2}$
23.	$y = 23\cos^2(x_1^3x_2^5) + 2x_1$	24.	$y = \lg_{10}(x_1x_2^2) + 45\sin(x_1 + x_2)$
25.	$y = \frac{\sqrt{\cos^2(x_1) + x_2}}{x_1^{13} + \frac{3}{\cos(x_2)}}$	26.	$y = \sqrt{\frac{x_2^2 + x_1/x_2}{16x_2x_1}}$
27.	$y = \sin(x_1 - x_2^3 + \sqrt{x_1}) - 1.3x_1^3$	28.	$y = \frac{\ln(x_2)}{\sqrt[3]{0.6x_1 \sin x_2 \cos x_1^4}}$
29.	$y = \frac{4\sin(3 + x_1x_2)}{34 - 9x_2^3}$	30.	$y = \cos^3\left(\frac{x_1 + 2x_2 + 9}{0.666}\right) + x_2^7$



Тема. Форматування виводу.

Мета. Ознайомлення принципами форматування рядків у C#; ознайомлення з описувачами стандартних форматів та рядками форматування стандартних числових форматів.

Завдання

Розробити консольний застосунок, який розраховуватиме заданий математичний вираз з двома змінними $f(x_1, x_2) = x_1^2 + e^{x_2}$ в певному діапазоні для обох змінних. Програма повинна забезпечувати ввід мінімальних $x1_{min}$, $x2_{min}$ максимальних $x1_{max}$, $x2_{max}$ значень, а також кроку зміни $dx1$, $dx2$. Всі розраховані значення виразу слід послідовно вивести на екран.

Виконання завдання

1. В середовищі Microsoft Visual Studio створити новий проект консольного застосунку. Для цього вибрати команду меню File/New/Project. У вікні New Project вибрати шаблон ConsoleApplication у розділі Templates/ Visual C#/ Windows. Далі вказати назву проекту (наприклад, Lab02) та його розташування на диску.
2. Продумати алгоритм програми. Алгоритм для виконання завдання такий:
 - 1) Ввести $x1Min$, $x1Max$, $dx1$, $x2Min$, $x2Max$, $dx2$.
 - 2) Тимчасовій змінній $x1$ присвоїти значення $x1Min$.
 - 3) У циклі, доки $x1 \leq x1Max$:
 - Тимчасовій змінній $x2$ присвоїти значення $x2Min$
 - У циклі, доки $x2 \leq x2Max$:
 - Розрахувати значення функції $y = f(x1, x2)$;
 - Вивести на екран: $x1$, $x2$ та y ;
 - Збільшити значення $x2$ на $dx2$;
 - Збільшити значення $x1$ на $dx1$;
3. У редакторі коду Microsoft Visual Studio у функцію Main() записати код з лістингу 2.1.
4. Запустити застосунок (F5 в середовищі Microsoft Visual Studio) і перевірити його роботу. Приклад виконання розробленої програми приведено на рис. 2.1.

Лістинг 2.1

```
Console.Write("Введіть початкове значення X1min: ");
string sx1Min = Console.ReadLine();
double x1Min = Double.Parse(sx1Min);

Console.Write("Введіть кінцеве значення X1max: ");
string sx1Max = Console.ReadLine();
double x1Max = double.Parse(sx1Max);

Console.Write("Введіть приріст dx1: ");
string sdx1 = Console.ReadLine();
double dx1 = double.Parse(sdx1);
```

```

Console.WriteLine("Введіть початкове значення X2min: ");
string sx2Min = Console.ReadLine();
double x2Min = Double.Parse(sx2Min);

Console.WriteLine("Введіть кінцеве значення X2max: ");
string sx2Max = Console.ReadLine();
double x2Max = double.Parse(sx2Max);

Console.WriteLine("Введіть приріст dx2: ");
string sdx2 = Console.ReadLine();
double dx2 = double.Parse(sdx2);

double y;

double x1 = x1Min;
double x2;

while (x1 <= x1Max)
{
    x2 = x2Min;
    while (x2 <= x2Max)
    {
        y = Math.Pow(x1, 2) + Math.Exp(x2);
        Console.WriteLine(
            "x1 = {0:###}\tx2 = {1:###}\ty = {2:###}", x1, x2, y);
        x2 += dx2;
    }
    x1 += dx1;
}

Console.ReadKey();

```

Національний університет
водного господарства
та природокористування

```

x1 = 34,135    x2 = 4,358    y = 1243,299
x1 = 34,135    x2 = 7,451    y = 2886,7821
x1 = 34,135    x2 = 10,544   y = 39114,2577
x1 = 34,135    x2 = 13,637   y = 837680,4232
x1 = 34,135    x2 = 16,73    y = 18448560,817
x1 = 34,135    x2 = 19,823   y = 406462758,0667
x1 = 34,135    x2 = 22,916   y = 8959678038,183
x1 = 34,135    x2 = 26,009   y = 197499127962,83
x1 = 34,135    x2 = 29,102   y = 4353494623659,66
x1 = 34,135    x2 = 32,195   y = 95964553036928,2
x1 = 34,135    x2 = 35,288   y = 2115357026462140
x1 = 34,135    x2 = 38,381   y = 46629043827572200
x1 = 35,082    x2 = 4,358    y = 1308,8475
x1 = 35,082    x2 = 7,451    y = 2952,3306
x1 = 35,082    x2 = 10,544   y = 39179,8062
x1 = 35,082    x2 = 13,637   y = 857745,9717
x1 = 35,082    x2 = 16,73    y = 18448626,3655
x1 = 35,082    x2 = 19,823   y = 406462823,6152
x1 = 35,082    x2 = 22,916   y = 8959678103,7315
x1 = 35,082    x2 = 26,009   y = 197499128028,379
x1 = 35,082    x2 = 29,102   y = 4353494623725,21
x1 = 35,082    x2 = 32,195   y = 95964553036991,8
x1 = 35,082    x2 = 35,288   y = 2115357026462210
x1 = 35,082    x2 = 38,381   y = 46629043827572200

```

Рис. 2.1. Вивід розробленого застосунку у консоль

Завдання для самостійного опрацювання

водного господарства

1. Вирішити розглянуте раніше завдання для заданої функції $f(x_1, x_2)$ згідно варіанту.

2. Додатково до попереднього завдання, обчислити та вивести на екран:

- для варіантів, номери яких закінчуються на 0 чи 5 – суму всіх додатних розрахованих проміжних значень $f(x_1, x_2)$;
- для варіантів, номери яких закінчуються на 1 чи 6 – добуток всіх від'ємних розрахованих проміжних значень $f(x_1, x_2)$;
- для варіантів, номери яких закінчуються на 2 чи 7 – суму додатних синусів всіх розрахованих проміжних значень $f(x_1, x_2)$;
- для варіантів, номери яких закінчуються на 3 чи 8 – добуток від'ємних косинусів всіх розрахованих проміжних значень $f(x_1, x_2)$;
- для варіантів, номери яких закінчуються на 4 чи 9 – суму додатних кубів всіх розрахованих проміжних значень $f(x_1, x_2)$;

3. При виводі числових значень на екран:

- для варіантів, номери яких закінчуються на 0...3 – виводити 4 знаки після коми;
- для варіантів, номери яких закінчуються на 4...6 – виводити 3 знаки після коми;
- для інших варіантів – виводити значення в експоненційному форматі.

Таблиця 2.1.

Варіант	Завдання	Варіант	Завдання
1.	$y = \frac{\sin^3(x_1) + 45 + x_2}{2x_1^4 + 4x_2}$	2.	$y = 0.1x_1 \sin x_2 \cos x_1^4 + 55$
3.	$y = \sqrt{\frac{x_2^2 + x_1/x_2}{\cos(x_1^3 x_2^5) + 2x_1}}$	4.	$y = \cos(\sqrt{x_2} + 34 \cdot \sin(x_1)) - 4 \sin(x_2)$
5.	$y = \frac{3x_2 - x_1^2}{\cos^3\left(\frac{x_1 + 2x_2 + 9}{0.37}\right)}$	6.	$y = \cos^3\left(\exp\left(\frac{x_1 + 2x_2 + 9}{0.666}\right)\right)$
7.	$y = \frac{\ln(x_2)}{\sqrt[3]{0.6x_1 \sin x_2 \cos x_1^4}}$	8.	$y = \sqrt{\frac{x_2^2 + x_1/x_2}{16x_2x_1}}$
9.	$y = 45 \sin(x_1 + x_2 + \lg_{10}(x_1x_2^2))$	10.	$y = \frac{\cos^2\left(\lg_{10} \frac{x_2}{x_1}\right)}{45 + x_2}$
11.	$y = \frac{5\sqrt{x_1^3 + x_2^5} - \cos(x_2)}{\exp(x_1)}$	12.	$y = \frac{\cos^3(x_1) + 45 + x_2}{x_1^{13} + \cos(x_2)}$
13.	$y = \frac{1}{4 + x_2} \cdot \sqrt{\cos^2\left(\frac{x_2}{x_1}\right)}$	14.	$y = 45x_1 \sin x_2 + \sqrt{9x_2x_1^3}$
15.	$y = \sqrt{56 + \frac{x_1 + x_2 + \sin(x_1x_2)}{5 \cos(x_2^2)}}$	16.	$y = 23 \sin^2(x_1^3 x_2^5) + 2x_1 + \cos(x_1x_2)$



Тема. Керування потоком інструкцій програми: умови та цикли.

Мета. Вивчення принципів використання інструкцій для керування потоком виконання програми.

Завдання

Розробити консольний застосунок, який розраховуватиме інтеграл неперервної функції з однією змінною $f(x)=x^2$ на заданому відрізку інтегрування $[a, b]$. Програма повинна забезпечувати ввід початку і кінця відрізка інтегрування a та b , а також кількості інтервалів інтегрування n . Результат обчислення слід вивести на екран.

Пояснення до завдання

Інтеграл функції можна уявити як площу фігури, обмеженої зверху графіком функції, знизу – віссю абсцис, а зліва і справа – вертикальними лініями, які проходять через краї відрізка інтегрування a та b (рис. 3.1).

Якщо цю фігуру розділити на n вертикальних смуг, то при $n \rightarrow \infty$ інтеграл дорівнюватиме сумі площ цих смуг:

$$S = \int_a^b f(x) \cdot dx = \lim_{n \rightarrow \infty} \sum_{i=1}^n f(x_i) \cdot \Delta x.$$

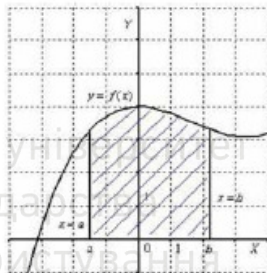


Рис. 3.1. Площа під кривою

При цьому є кілька підходів до формування цих елементарних смуг, які відрізняються способом вибору двох верхніх точок (дві нижні точки завжди розташовані на осі абсцис).

Перший підхід передбачає, що елементарні смуги є прямокутними. Його називають *методом прямокутників*. Для чисельного обчислювання інтегралів методом прямокутників проміжок інтегрування $[a, b]$ поділяють на n рівних частин. Задану функцію $f(x)$ на кожному відрізку замінюють на прямі лінії, паралельні до осі абсцис. При цьому криволінійна функція замінюється на n прямокутників. Інтеграл розраховують як суму площ n прямокутників, отриманих шляхом розбиття відрізка інтегрування $[a, b]$ на n рівних частин. Чим більша кількість ділянок n , на які ділимо інтервал $[a, b]$, тим точніше сукупність прямокутників відтворює функцію, і тим точнішим буде інтеграл.

Є три підвиди методу прямокутників, які визначають, як саме формуватиметься верхня сторона кожної ділянки:

- 1) *Метод лівих прямокутників*. Верхня сторона ділянки проходить через точку перетину лівої сторони ділянки з функцією (рис. 3.2, а).

- 2) Метод *правих* прямокутників. Верхня сторона ділянки проходить через точку перетину правої сторони ділянки з функцією (рис. 3.2, б).
- 3) Метод *центральных* прямокутників. Верхня сторона ділянки проходить через точку перетину вертикальної осі ділянки з функцією (рис. 3.2, в).

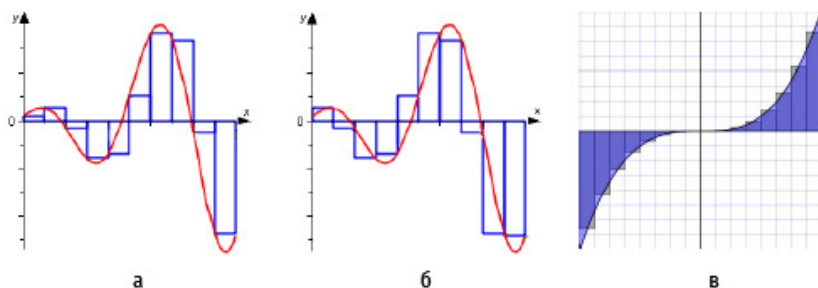


Рис.3.2. Ілюстрація методу лівих прямокутників (а), правих прямокутників (б), та центральных прямокутників (в)

Ще один підхід до формування елементарних смуг передбачає, що вони мають форму трапеції. Його називають *методом трапецій*. Фактично цей підхід полягає у заміні кривої підінтегральної функції на ламану. Проміжок $[a, b]$ розбивають на n рівних частин, та сполучають прямими лініями значення функції на кінцях відрізків, тобто, площу криволінійної функції наближено замінюємо на суму площ n трапецій (рис. 3.3). Чим більша кількість ділянок n , на які розділено інтервал $[a, b]$, тим точніше сукупність трапецій відтворює функцію, і тим точнішим буде інтеграл.

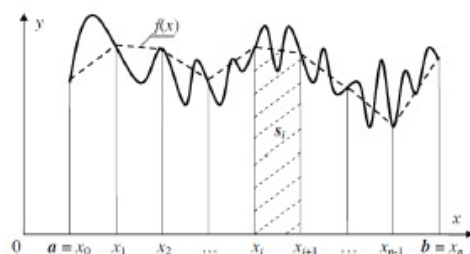


Рис. 3.3. Ілюстрація методу трапецій

Виконання завдання

- В середовищі Microsoft Visual Studio створити новий проект консольного застосунку. Для цього вибрати команду меню File/New/Project. У вікні New Project вибрати шаблон ConsoleApplication у розділі Templates/ Visual C#/ Windows. Далі вказати назву проекту (наприклад, Lab03) та його розташування на диску.
- Продумати алгоритм програми. Алгоритм для виконання завдання такий:



- 1) Ввести a, b, n .
- 2) Розрахувати величину кроку інтегрування dx .
- 3) Тимчасовій змінній *Intgrl* присвоїти значення 0.
- 4) У циклі, для $i = 0 \dots n-1$:
 - Розрахувати початкову та кінцеву межу i -ої поточної ділянки;
 - Розрахувати значення функції $f(x)$ на межах поточної ділянки;
 - Розрахувати площу поточної ділянки і додати її до змінної *Intgrl*;
- 5) Вивести на екран результат розрахунку інтегралу;
- 6) Перепитати користувача, чи здійснити ще один розрахунок
 - Якщо так, перейти до п.1;
3. У редакторі коду Microsoft Visual Studio в класі Program (безпосередньо перед функцією Main()) записати код з лістингу 3.1.
4. У функцію Main() записати код з лістингу 3.2.
5. Запустити застосунок (F5 в середовищі Microsoft Visual Studio) і перевірити його роботу. Приклад виконання розробленої програми приведено на рис. 3.4.

Лістинг 3.1

```
static double Function(double x)
{
    return x * x;
}
```

Лістинг 3.2

```
StartOfCalculations:
Console.WriteLine("Введіть початок відрізка інтегрування a: ");
string sa = Console.ReadLine();
double a = double.Parse(sa);

Console.WriteLine("Введіть кінець відрізка інтегрування b: ");
string sb = Console.ReadLine();
double b = double.Parse(sb);

Console.WriteLine("Введіть кількість ділянок n: ");
string sn = Console.ReadLine();
double n = double.Parse(sn);

double dx = (b - a) / n;
double y1, y2;
double x1, x2;
double Intgrl = 0;

for (int i = 0; i < n; i++)
{
    x1 = a + i * dx;
    x2 = x1 + dx;
    y1 = Function(x1);
    y2 = Function(x2);

    // Обчислення інтегралу методом центральних прямокутників
    Intgrl += (y1 + y2) / 2 * dx;
}

Console.WriteLine(
    "Інтеграл функції на відрізку [{0}, {1}] становить {2:0.0000}", a, b,
    Intgrl);
```

```

Console.WriteLine("Повторити розрахунок (у - так) ? ");
ConsoleKeyInfo pressedKey = Console.ReadKey();
Console.WriteLine();
if (pressedKey.Key == ConsoleKey.Y)
{
    Console.WriteLine();
    goto StartOfCalculations;
}

```

```

New/D/Programs/MSVS/C#/Lab03/Lab03/bin/Debug/Lab03.ESE
Введіть початок відрізка інтегрування a: 1
Введіть кінець відрізка інтегрування b: 11
Введіть кількість ділянок n: 10
Інтеграл функції на відріжку [1, 11] становить 445
Повторити розрахунок (у - так) ? у

Введіть початок відрізка інтегрування a: 1
Введіть кінець відрізка інтегрування b: 11
Введіть кількість ділянок n: 50
Інтеграл функції на відріжку [1, 11] становить 443,4
Повторити розрахунок (у - так) ? у

Введіть початок відрізка інтегрування a: 1
Введіть кінець відрізка інтегрування b: 11
Введіть кількість ділянок n: 100
Інтеграл функції на відріжку [1, 11] становить 443,35
Повторити розрахунок (у - так) ? у

Введіть початок відрізка інтегрування a: 1
Введіть кінець відрізка інтегрування b: 11
Введіть кількість ділянок n: 1000
Інтеграл функції на відріжку [1, 11] становить 443,3335
Повторити розрахунок (у - так) ? у

Введіть початок відрізка інтегрування a: 1

```

Рис. 3.4. Вивід розробленого застосунку у консоль

Завдання для самостійного опрацювання

- Вирішити розглянуте раніше завдання для заданої функції $f(x)$ згідно варіанту. У виразах прийняти, що $x_1=2x$, $x_2=5x$. При цьому:
 - для варіантів, номери яких закінчуються на 0...3 – використати метод лівих прямокутників; при виводі результату виводити 4 знаки після коми;
 - для варіантів, номери яких закінчуються на 4...6 – використати метод правих прямокутників; при виводі результату виводити 5 знаків після коми;
 - для варіантів, номери яких закінчуються на 7...9 – використати метод трапецій; при виводі результату виводити 6 знаків після коми;



Варіант	Завдання	Варіант	Завдання
1.	$y = \frac{\sqrt{x_1^3 + x_2^5}}{1000} + 65$	2.	$y = \cos(x_1 - x_2^2) + 31.55x_2x_1^2$
3.	$y = \frac{6 - \cos(3 + x_1)}{34 - 9x_2^3 + x_2}$	4.	$y = \cos^4\left(x_1 - \sqrt{\frac{x_2}{x_1 + 53x_2^2}}\right)$
5.	$y = 23 \sin^2(x_1^3 x_2^5) + 2x_1 + \cos(x_1 x_2)$	6.	$y = \sqrt{56 + \frac{x_1 + x_2 + \sin(x_1 x_2)}{5 \cos(x_2^2)}}$
7.	$y = 45x_1 \sin x_2 + \sqrt{9x_2 x_1^3}$	8.	$y = \frac{1}{4 + x_2} \cdot \sqrt{\cos^2\left(\exp \frac{x_2}{x_1}\right)}$
9.	$y = \frac{\cos^3(x_1) + 45 + x_2}{x_1^{13} + \cos(x_2)}$	10.	$y = \frac{5\sqrt{x_1^3 + x_2^5} - \cos(x_2)}{\sin(x_1)}$
11.	$y = \frac{\sqrt{x_1^3 + x_2^5}}{1000 \cdot \sqrt{x_1 + x_2^5}} + 65$	12.	$y = 23 \cos^2(x_1^3 x_2^5) + 2x_1$
13.	$y = \frac{\sqrt{\cos^3(x_1) + x_2}}{x_1^{13} + \sqrt[3]{\cos(x_2)}}$	14.	$y = \sin(x_1 - x_2^3 + \sqrt{x_1}) - 1.3x_1^3$
15.	$y = \frac{4 \sin(3 + x_1 x_2)}{34 - 9x_2^3 + x_2}$	16.	$y = \sqrt{56x_1 + \frac{x_1 + x_2 + \sin(x_1 x_2)}{5 - \cos(x_2^2)}}$
17.	$y = \sqrt[5]{0.1x_1 \sin x_2 \cos x_1^2 + 55x_1 x_2}$	18.	$y = \sqrt{\frac{\cos(2x_2) + x_1/x_2}{16x_2 x_1}}$
19.	$y = \cos(\sqrt{x_2} + 34x_1) - 4 \sin(x_2)$	20.	$y = \sin^2\left(x_1 \frac{x_2}{x_1 + 53x_2^2}\right)$
21.	$y = \frac{3x_2 - x_1^2}{\cos^3\left(\exp\left(\frac{x_1 + 2x_2 + 9}{0.37}\right)\right)}$	22.	$y = \cos(\sqrt{x_2} + 34 \cdot \sin(x_1)) - 4 \sin(x_2)$
23.	$y = \sqrt{\frac{x_2^2 + x_1/x_2}{\cos(x_1^3 x_2^5) + 2x_1}}$	24.	$y = 0.1x_1 \sin x_2 \cos x_1^4 + 55$
25.	$y = \frac{\sin^3(x_1) + 45 + x_2}{2x_1^4 + 4x_2}$	26.	$y = \frac{\cos^2\left(\lg_{10} \frac{x_2}{x_1}\right)}{45 + x_2}$
27.	$y = \lg_{10}(x_1 x_2^2) + 45 \sin(x_1 + x_2)$	28.	$y = \sqrt{\frac{x_2^2 + x_1/x_2}{16x_2 x_1}}$
29.	$y = \frac{\ln(x_2)}{\sqrt[5]{0.6x_1 \sin x_2 \cos x_1^4}}$	30.	$y = \cos^3\left(\frac{x_1 + 2x_2 + 9}{0.666}\right) + x_2^7$



Тема. Робота з масивом.

Мета. Вивчення засобів C# для роботи з масивами.

Завдання

Розробити консольний застосунок, який створить масив та заповнить його значеннями функції $f(x) = x^2$ в певному діапазоні. Програма повинна сортувати масив за спаданням значень та розрахувати мінімальне $aMin$, максимальне $aMax$ та середнє $aAvg$ значення масиву. Відсортований масив та знайдені $aMin$, $aMax$, $aAvg$ слід вивести на екран.

Виконання завдання

1. В середовищі Microsoft Visual Studio створити новий проект консольного застосунку. Для цього вибрати команду меню File/New/Project. У вікні New Project вибрати шаблон ConsoleApplication у розділі Templates/ Visual C#/ Windows. Далі вказати назву проекту (наприклад, Lab04) та його розташування на диску.
2. Продумати алгоритм програми. Алгоритм для виконання завдання такий:
 - 1) Оголосити і створити масив arr , заповнити його значеннями функції $y = f(x)$.
 - 2) Відсортувати масив arr за спаданням значень.
 - 3) Вивести масив arr на екран.
 - 4) Знайти мінімальне $aMin$, максимальне $aMax$ та середнє $aAvg$ значення масиву.
 - 5) Вивести знайдені значення $aMin$, $aMax$, $aAvg$ на екран.
3. У класі Program оголосити константи та метод для розрахунку функції (лістинг 4.1).
4. У функцію Main() записати код з лістингу 4.2.
5. Запустити застосунок (F5 в середовищі Microsoft Visual Studio) і перевірити його роботу. Приклад виконання розробленої програми приведено на рис. 4.1.

Лістинг 4.1

```
const double StartX = 10.3;  
const double dX = 0.7;  
  
static double Function(double x)  
{  
    return x * x;  
}
```

Лістинг 4.2

```
double[] arr = new double[10];  
  
double x = StartX;  
for (int i = arr.GetLowerBound(0); i <= arr.GetUpperBound(0); i++)  
{
```


arr[i] = Function(x);
x += dx;
}

```
Array.Sort(arr);
Array.Reverse(arr);

Console.WriteLine("Відсортовані за спаданням значення масиву: ");

for (int i = arr.GetLowerBound(0); i <= arr.GetUpperBound(0); i++)
{
    Console.WriteLine("arr[{0:00}] = {1:0.0000}", i, arr[i]);
}

double aMin = arr[arr.GetUpperBound(0)];
double aMax = arr[arr.GetLowerBound(0)];

double aAvg = 0;
for (int i = arr.GetLowerBound(0); i <= arr.GetUpperBound(0); i++)
{
    aAvg += arr[i];
}
aAvg = aAvg / arr.GetLength(0);

Console.WriteLine("Мінімальне значення масиву: {0:0.0000}", aMin);
Console.WriteLine("Максимальне значення масиву: {0:0.0000}", aMax);
Console.WriteLine("Середнє значення масиву: {0:0.0000}", aAvg);

Console.ReadKey(true);
```

```
File:///C:/Program%20Files/C#/Lib04/Lib04/In/Dev/Lib04/E
Відсортовані за спаданням значення масиву:
arr[00] = 275,5600
arr[01] = 252,8100
arr[02] = 231,0400
arr[03] = 210,2500
arr[04] = 190,4400
arr[05] = 171,6100
arr[06] = 153,7600
arr[07] = 136,8900
arr[08] = 121,0000
arr[09] = 106,0900
Мінімальне значення масиву: 106,0900
Максимальне значення масиву: 275,5600
Середнє значення масиву: 184,9450
```

Рис. 4.1. Вивід розробленого застосунку у консоль

Завдання для самостійного опрацювання

1. Вирішити розглянуте раніше завдання для заданої функції $f(x)$ згідно варіанту. У виразах прийняти, що $x_1=2.76x$, $x_2=0.5x$.
2. Додатково до попереднього завдання, обчислити та вивести на екран значення R згідно варіанту.

Варіант	Завдання	Варіант	Завдання
1.	$y = \frac{3x_2 - x_1^2}{\cos^3\left(\exp\left(\frac{x_1 + 2x_2 + 9}{0.37}\right)\right)}$	2.	$y = \cos(\sqrt{x_2} + 34 \cdot \sin(x_1)) - 4 \sin(x_2)$
3.	$y = \sqrt{\frac{x_2^2 + x_1/x_2}{\cos(x_1^3 x_2^5) + 2x_1}}$	4.	$y = 0.1x_1 \sin x_2 \cos x_1^4 + 55$
5.	$y = \frac{\sin^3(x_1) + 45 + x_2}{2x_1^4 + 4x_2}$	6.	$y = \frac{\cos^2\left(\lg_{10} \frac{x_2}{x_1}\right)}{45 + x_2}$
7.	$y = \lg_{10}(x_1 x_2^2) + 45 \sin(x_1 + x_2)$	8.	$y = \sqrt{\frac{x_2^2 + x_1/x_2}{16x_2 x_1}}$
9.	$y = \frac{\ln(x_2)}{\sqrt[3]{0.6x_1 \sin x_2 \cos x_1^4}}$	10.	$y = \cos^3\left(\frac{x_1 + 2x_2 + 9}{0.666}\right) + x_2^7$
11.	$y = \sqrt[3]{56x_1 + \frac{x_1 + x_2 + \sin(x_1 x_2)}{5 - \cos(x_2^2)}}$	12.	$y = \sqrt[3]{0.1x_1 \sin x_2 \cos x_1^2 + 55x_1 x_2}$
13.	$y = \sqrt{\frac{\cos(2x_2) + x_1/x_2}{16x_2 x_1}}$	14.	$y = \cos(\sqrt{x_2} + 34x_1) - 4 \sin(x_2)$
15.	$y = \sin^2\left(x_1 \frac{x_2}{x_1 + 53x_2^2}\right)$	16.	$y = \frac{\sqrt{x_1^3 + x_2^5}}{1000 \cdot \sqrt{x_1 + x_2^5}} + 65$
17.	$y = 23 \cos^2(x_1^3 x_2^5) + 2x_1$	18.	$y = \frac{\sqrt{\cos^3(x_1) + x_2}}{x_1^{13} + \frac{3}{\cos(x_2)}}$
19.	$y = \sin(x_1 - x_2^3 + \sqrt{x_1}) - 1.3x_1^3$	20.	$y = \frac{4 \sin(3 + x_1 x_2)}{34 - 9x_2^3}$
21.	$y = \frac{\sqrt{x_1^3 + x_2^5}}{1000} + 65$	22.	$y = \cos(x_1 - x_2^2) + 31.55x_2 x_1^2$
23.	$y = \frac{6 - \cos(3 + x_1)}{34 - 9x_2^3 + x_2}$	24.	$y = \cos^4\left(x_1 - \sqrt{\frac{x_2}{x_1 + 53x_2^2}}\right)$
25.	$y = 23 \sin^2(x_1^3 x_2^5) + 2x_1 + \cos(x_1 x_2)$	26.	$y = \sqrt[3]{56 + \frac{x_1 + x_2 + \sin(x_1 x_2)}{5 \cos(x_2^2)}}$
27.	$y = 45x_1 \sin x_2 + \sqrt{9x_2 x_1^3}$	28.	$y = \frac{1}{4 + x_2} \cdot \sqrt{\cos^2\left(\exp \frac{x_2}{x_1}\right)}$
29.	$y = \frac{\cos^3(x_1) + 45 + x_2}{x_1^{13} + \cos(x_2)}$	30.	$y = \frac{5\sqrt{x_1^3 + x_2^5} - \cos(x_2)}{\sin(x_1)}$



Остання цифра R номера варіанту	
0	Кількість елементів масиву, менших за середнє значення
1	Сума елементів масиву, менших за середнє значення
2	Кількість елементів масиву, більших за середнє значення
3	Сума елементів масиву, більших за середнє значення
4	Кількість елементів масиву, які знаходяться в діапазоні $aMin...aMin+10\%aMin$
5	Сума елементів масиву, які знаходяться в діапазоні $aMin...aMin+10\%aMin$
6	Кількість елементів масиву, які знаходяться в діапазоні $aMax-10\%aMax...aMax$
7	Сума елементів масиву, які знаходяться в діапазоні $aMax-10\%aMax...aMax$
8	Кількість елементів масиву, які знаходяться в діапазоні $aAvg-10\%aAvg...aAvg+10\%aAvg$
9	Суму елементів масиву, які знаходяться в діапазоні $aAvg-10\%aAvg...aAvg+10\%aAvg$





Тема. Проектування простого класу.

Мета. Вивчення принципів розробки та використання класів; вивчення принципів роботи з об'єктами.

Завдання

Розробити консольний застосунок, у якому описати клас, що представляє об'єкт "місто". Клас має містити не менше 7 полів різних типів і один метод. Програма має забезпечити ввід даних для одного екземпляру, створити об'єкт розробленого класу, провести дії, представлені розробленим методом класу, і вивести результати на екран.

Виконання завдання

1. В середовищі Microsoft Visual Studio створити новий проект консольного застосунку. Для цього вибрати команду меню File/New/Project. У вікні New Project вибрати шаблон ConsoleApplication у розділі Templates/ Visual C#/ Windows. Далі вказати назву проекту (наприклад, Lab05) та його розташування на диску.
2. Продумати алгоритм програми. Для цього слід:
 - 1) Проаналізувати предметну область. Потрібно виділити важливі властивості об'єкта – вони будуть представлені полями. Виявити, яку дію слід виконати над даними об'єкта – вона буде реалізована методом класу. Скласти перелік полів (з типами) та методів.
 - 2) Описати у програмі клас (поля, методи та реалізація методів).
 - 3) Далі у програмі слід:
 - Скласти програмні інструкції для введення даних, які будуть записані у поля об'єкта – екземпляра класу.
 - Створити об'єкт – екземпляр розробленого класу.
 - Записати введені дані у відповідні поля об'єкта.
 - Виконати метод, який обробляє дані об'єкта
 - Вивести на екран всі поля об'єкта та результат роботи метода.
3. Для заданої у завданні предметної області ("місто") вважаємо важливими такі "властивості":
 - Назва міста
 - Назва країни
 - Назва регіону (області, провінції)
 - Населення (кількість)
 - Річний дохід міста
 - Площа
 - Чи має місто порт



- Чи має місто аеропорт

Над даними об'єкта слід виконати таку дію:

- Обчислити середній річний дохід, який припадає на одного мешканця
4. У просторі імен проекту записати оголошення класу з лістингу 5.1.
 5. У функцію Main() записати код з лістингу 5.2.
 6. Запустити застосунок (F5 в середовищі Microsoft Visual Studio) і перевірити його роботу.
- Приклад виконання розробленої програми приведено на рис. 5.1.

Лістинг 5.1

```
class Town
{
    public string Name;
    public string Country;
    public string Region;
    public int Population;
    public double YearIncome;
    public double Square;
    public bool HasPort;
    public bool HasAirport;

    public double GetYearIncomePerInhabitant()
    {
        return YearIncome / Population;
    }
}
```

Лістинг 5.2

```
Console.WriteLine("Введіть назву міста: ");
string sName = Console.ReadLine();

Console.WriteLine("Введіть назву країни: ");
string sCountry = Console.ReadLine();

Console.WriteLine("Введіть назву регіону: ");
string sRegion = Console.ReadLine();

Console.WriteLine("Введіть кількість населення: ");
string sPopulation = Console.ReadLine();

Console.WriteLine("Введіть річний дохід: ");
string sYearIncome = Console.ReadLine();

Console.WriteLine("Введіть площу, кв. км: ");
string sSquare = Console.ReadLine();

Console.WriteLine("Чи є у місті порт? (у-так, н-ні): ");
ConsoleKeyInfo keyHasPort = Console.ReadKey();
Console.WriteLine();

Console.WriteLine("Чи є у місті аеропорт? (у-так, н-ні): ");
ConsoleKeyInfo keyHasAirport = Console.ReadKey();
Console.WriteLine();

Town OurTown = new Town();

OurTown.Name = sName;
OurTown.Country = sCountry;
OurTown.Region = sRegion;
OurTown.Population = int.Parse(sPopulation);
OurTown.YearIncome = double.Parse(sYearIncome);
OurTown.Square = double.Parse(sSquare);
OurTown.HasPort = keyHasPort.Key == ConsoleKey.Y ? true : false;
OurTown.HasAirport = keyHasAirport.Key == ConsoleKey.Y ? true : false;
```

```
double YearIncomePerInhabitant = OurTown.GetYearIncomePerInhabitant();
Console.WriteLine();
Console.WriteLine("-----");
Console.WriteLine("Дані про об'єкт: ");
Console.WriteLine("-----");
Console.WriteLine("Назва: " + OurTown.Name);
Console.WriteLine("Країна: " + OurTown.Country);
Console.WriteLine("Регіон: " + OurTown.Region);
Console.WriteLine("Кількість населення: " +
    OurTown.Population.ToString());
Console.WriteLine("Річний дохід: " +
    OurTown.YearIncome.ToString("0.00"));
Console.WriteLine("Площа: " + OurTown.Square.ToString("0.000"));
Console.WriteLine(OurTown.HasPort ? "У місті є порт" :
    "У місті нема порту");
Console.WriteLine(OurTown.HasAirport ? "У місті є аеропорт" :
    "У місті нема аеропорту");
Console.WriteLine();
Console.WriteLine("Середній річний дохід на одного громадянина: " +
    YearIncomePerInhabitant.ToString("0.00"));
Console.ReadKey();
```

```
File://C:/Program/CSharp/MSVS/CA/Lab01/Lab01/bin/Debug/Lab01.E11
Введіть назву міста: Херсон
Введіть назву країни: Україна
Введіть назву регіону: Херсонська область
Введіть кількість населення: 296448
Введіть річний дохід: 1000000000
Введіть площу, кв. км: 145
Чи є у місті порт? (у-так, н-ні): у
Чи є у місті аеропорт? (у-так, н-ні): у

-----
Дані про об'єкт:
-----
Назва: Херсон
Країна: Україна
Регіон: Херсонська область
Кількість населення: 296448
Річний дохід: 1000000000,00
Площа: 145,000
У місті є порт
У місті є аеропорт

Середній річний дохід на одного громадянина: 3373,27
```

Рис. 5.1. Вивід розробленого застосунку у консоль

Завдання для самостійного опрацювання

Вирішити розглянуте раніше завдання щодо заданої згідно варіанту предметної області. Клас має містити не менше 7 полів та не менше одного методу. Характеристики об'єкта вибрати самостійно, довільно, відповідно до предметної області.



Варіант	Завдання	Варіант	Завдання
1.	Комп'ютер	2.	Автомобіль
3.	Принтер	4.	Літак
5.	Фотоапарат	6.	Процесор
7.	Телефон	8.	Телевізор
9.	Велосипед	10.	Планшет
11.	Абонент телефонної мережі	12.	Книга
13.	Студент	14.	Персона
15.	Документ (паспорт)	16.	Будівля
17.	Планета	18.	Футбольна команда
19.	Рослина	20.	Тварина
21.	Жорсткі диски до ЕОМ	22.	Країна
23.	Область	24.	Ріка
25.	Озеро	26.	Завод
27.	Університет	28.	Фільм
29.	Музичний виконавець	30.	Ферма





Тема. Розробка бібліотеки класів.

Мета. Вивчення принципів розробки та використання бібліотеки класів; вивчення принципів роботи з масивами та об'єктами; ознайомлення з принципами формування рішень з кількома проектами.

Завдання

Розробити бібліотеку з класом, створеним на основі класу з лабораторної роботи №5. Клас доповнити однією властивістю.

Розробити консольний застосунок, який використовує клас з бібліотеки. Програма має забезпечити ввід даних для масиву об'єктів спроектованого класу, і вивести результати для всіх об'єктів масиву на екран.

Виконання завдання

Розробка бібліотеки

1. В середовищі Microsoft Visual Studio створити новий проект бібліотеки класів. Для цього вибрати команду меню File/New/Project. У вікні New Project вибрати шаблон ClassLibrary у розділі Templates/ Visual C#/ Windows. Далі вказати назву проекту (наприклад, Lab06Lib) та його розташування на диску.
2. Дати назву файлу, в якому описуватимемо клас. Для цього у вікні Solution Explorer викликати контекстне меню на елементі Class1.cs і вибрати команду Rename. Задати назву файлу відповідно до його призначення (наприклад, ClassesLib).
3. У файлі ClassesLib.cs замість автоматично створеного класу Class1 ввести код свого класу з лабораторної роботи №5. Доповнити клас однією властивістю, спроектувавши її відповідно до предметної області. Повний код класу для об'єкта типу "місто" приведено у лістингу 6.1. Клас доповнено властивістю yearIncomePerInhabitant (лише для читання), яка повертає дані, обчислені методом GetYearIncomePerInhabitant() на основі даних об'єкта.
4. Скомпільовати бібліотеку. Для цього у вікні Solution Explorer викликати контекстне меню на елементі проекту бібліотеки (Lab06Lib) та вибрати команду Build. Якщо код не містить синтаксичних помилок, у відповідній папці, де збережено проект бібліотеки, буде створено файл бібліотеки з розширенням dll (в нашому випадку – Lab06Lib.dll). На цьому розробка бібліотеки закінчена.

Розробка застосунку, який використовуватиме бібліотеку

1. У поточному рішенні створити новий проект консольного застосунку. Для цього у вікні Solution Explorer викликати контекстне меню на елементі рішення (Solution 'Lab06Lib') і вибрати команду Add/ New Project. У вікні Add New Project вибрати шаблон

ConsoleApplication у розділі Templates/ Visual C#/ Windows. Дали вказати назву проекту (наприклад, Lab06App) та його розташування на диску. Після створення нового проекту у вікні Solution Explorer будуть присутні два проекти: розроблений перед цим Lab06Lib та щойно створений Lab06App (рис. 6.1).

- Щоб в новому проекті можна було використовувати типи з бібліотеки, її слід додати до посилань (references) проекту. Для цього у вікні Solution Explorer викликати контекстне меню на елементі References нового проекту (Lab06App) і вибрати команду Add Reference. У вікні Reference Manager (рис. 6.2) перейти у розділ Projects/ Solution. У центральній частині вікна буде перелік доступних посилань на збірки поточного рішення. В нашому випадку буде лише одне посилання – на попередню розроблену бібліотеку Lab06Lib. Слід відмітити його маркером і натиснути кнопку OK.

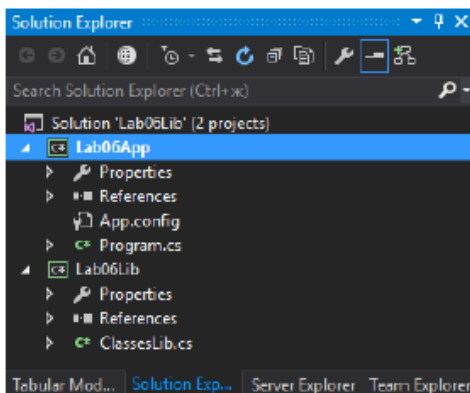


Рис. 6.1.

Зауважимо, що додати посилання на бібліотеку можна також безпосередньо вказавши її файл. Наприклад, замість дій, описаних в цьому пункті, у вікні Reference Manager можна натиснути кнопку Browse і вибрати файл Lab06Lib.dll, скомпільований при розробці бібліотеки. Так роблять, якщо проект бібліотеки не входить до складу поточного рішення.

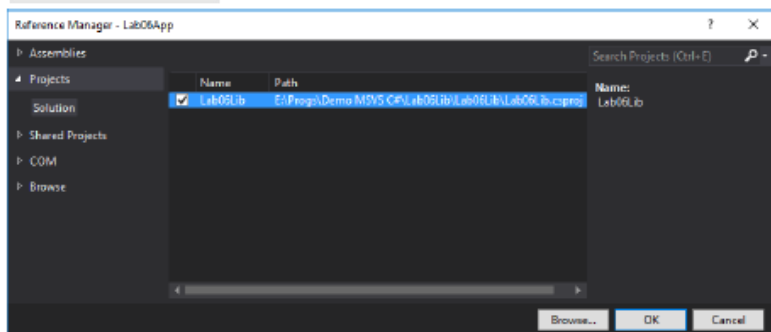


Рис. 6.2. Додання нового посилання

- Щоб у коді застосунку можна було використовувати типи з під'єднаної бібліотеки за їх скороченими іменами, в верхню частину файла Program.cs слід додати рядок

using Lab06Lib;

Тут Lab06Lib – простір імен у проекті бібліотеки (див. лістинг 6.1).

- Продумати алгоритм програми. Алгоритм для виконання завдання такий:



```
file:///E:/Progs/Demo/MSVS C#/Lab06Lib/Lab06App/bin/Debug/Lat
Введіть кількість міст: 2
Введіть назву міста: Тернопіль
Введіть назву країни: Україна
Введіть назву регіону: Тернопільська область
Введіть кількість населення: 230000
Введіть річний дохід: 12000000
Введіть площу, кв. км: 340000
Чи є у місті порт? (у-так, н-ні): н
Чи є у місті аеропорт? (у-так, н-ні): у

Введіть назву міста: Херсон
Введіть назву країни: Україна
Введіть назву регіону: Херсонська область
Введіть кількість населення: 380000
Введіть річний дохід: 14000000
Введіть площу, кв. км: 400000
Чи є у місті порт? (у-так, н-ні): у
Чи є у місті аеропорт? (у-так, н-ні): у

-----
Дані про місто Тернопіль
-----
Країна: Україна
Регіон: Тернопільська область
Кількість населення: 230000
Річний дохід: 12000000,00
Площа: 340000,000
У місті нема порту
У місті є аеропорт
```

Завдання для самостійного опрацювання

Вирішити розглянуте раніше завдання щодо заданої згідно варіанту предметної області (див. завдання до лабораторної роботи №5 – табл. 5.1).

Рис. 6.3. Вивід розробленого застосунку у консоль

```
public class Town
{
    public string Name;
    public string Country;
    public string Region;
    public int Population;
    public double YearIncome;
    public double Square;
    public bool HasPort;
    public bool HasAirport;

    public double yearIncomePerInhabitant
    {
        get
        {
            return GetYearIncomePerInhabitant();
        }
    }

    public double GetYearIncomePerInhabitant()
    {
        return YearIncome / Population;
    }
}
```

Лістинг 6.1

```
Town[] arrTowns;

Console.WriteLine("Введіть кількість міст: ");
int cntTowns = int.Parse(Console.ReadLine());
arrTowns = new Town[cntTowns];

for (int i = 0; i < cntTowns; i++)
{
    Console.WriteLine("Введіть назву міста: ");
```

Лістинг 6.2

```

string sName = Console.ReadLine();
Console.WriteLine("Введіть назву країни: ");
string sCountry = Console.ReadLine();

Console.WriteLine("Введіть назву регіону: ");
string sRegion = Console.ReadLine();

Console.WriteLine("Введіть кількість населення: ");
string sPopulation = Console.ReadLine();

Console.WriteLine("Введіть річний дохід: ");
string sYearIncome = Console.ReadLine();

Console.WriteLine("Введіть площу, кв. км: ");
string sSquare = Console.ReadLine();

Console.WriteLine("Чи є у місті порт? (у-так, н-ні): ");
ConsoleKeyInfo keyHasPort = Console.ReadKey();
Console.WriteLine();

Console.WriteLine("Чи є у місті аеропорт? (у-так, н-ні): ");
ConsoleKeyInfo keyHasAirport = Console.ReadKey();
Console.WriteLine();

Console.WriteLine();

Town theTown = new Town();
theTown.Name = sName;
theTown.Country = sCountry;
theTown.Region = sRegion;
theTown.Population = int.Parse(sPopulation);
theTown.YearIncome = double.Parse(sYearIncome);
theTown.Square = double.Parse(sSquare);
theTown.HasPort = keyHasPort.Key == ConsoleKey.Y ? true : false;
theTown.HasAirport = keyHasAirport.Key == ConsoleKey.Y ? true : false;

arrTowns[i] = theTown;
}

foreach (Town t in arrTowns)
{
    Console.WriteLine();
    Console.WriteLine("-----");
    Console.WriteLine("Дані про місто {0}", t.Name);
    Console.WriteLine("-----");
    Console.WriteLine("Країна: " + t.Country);
    Console.WriteLine("Регіон: " + t.Region);
    Console.WriteLine("Кількість населення: " +
        t.Population.ToString());
    Console.WriteLine("Річний дохід: " +
        t.YearIncome.ToString("0.00"));
    Console.WriteLine("Площа: " + t.Square.ToString("0.000"));
    Console.WriteLine(t.HasPort ? "У місті є порт" :
        "У місті нема порту");
    Console.WriteLine(t.HasAirport ? "У місті є аеропорт" :
        "У місті нема аеропорту");
    Console.WriteLine();
    Console.WriteLine("Середній річний дохід на одного громадянина: " +
        t.yearIncomePerInhabitant.ToString("0.00"));
}

Console.ReadKey();

```



Лабораторна робота №7

Тема. Розробка простого застосунку Windows Forms.

Мета. Вивчення принципів розробки застосунків з графічним інтерфейсом користувача на основі технології Windows Forms.

Завдання

Розробити одновіконний застосунок на основі технології Windows Forms для розрахунку арифметичного виразу на основі функції з двома змінними $f(x_1, x_2)$. Програма повинна забезпечувати ввід значень x_1 та x_2 , розрахунок значення функції $f(x_1, x_2) = x_1 \cdot x_2$, вивід результату розрахунку у вікно та очищення полів для наступного розрахунку.

Виконання завдання

В середовищі Microsoft Visual Studio створити новий проект застосунку Windows Forms. Для цього вибрати команду меню File/New/Project. У вікні New Project вибрати шаблон Windows Forms Application у розділі Templates/ Visual C#/ Windows. Далі вказати назву проекту (наприклад, Lab07) та його розташування на диску. Буде створено проект з однією порожньою формою (рис. 7.1).

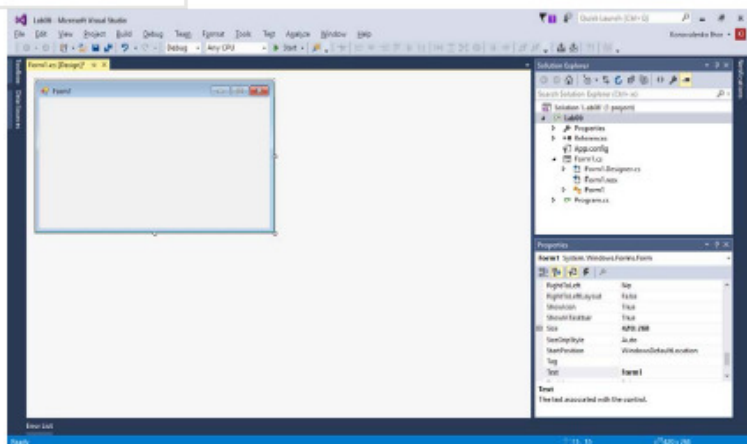


Рис. 7.1. Новий проект з порожньою формою

Вікно застосунку проектується відповідно до його цілей. Для виконання нашого завдання вікно повинно мати поля для вводу значень x_1 та x_2 та поле для виводу результату розрахунку функції $f(x_1, x_2)$. Крім цього, інтерфейс має містити засоби для ініціювання потрібних дій (розрахунок, очищення форми, вихід). Загальний вигляд вікна застосунку, розробка якого описана далі, показано на рис. 7.2.

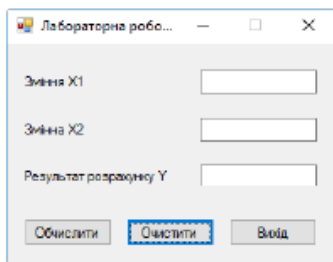


Рис. 7.2. Вікно застосунку

1. Змінити назву файлу, який описує форму (Form1.cs), та назву самої форми, на fMain¹. Для цього у вікні Solution Explorer викликати контекстне меню на елементі Form1.cs та вибрати команду Rename. Перейменувати файл у fMain.cs. При цьому середовище перепитає, чи змінити у проєкті також і змінну Form1, яка представляє форму. Вибрати "Так". Якщо вибрати "Ні", то назву форми слід змінити вручну. Для цього в дизайнері форм клікнути один раз на формі, далі у вікні Properties знайти властивість Name та задати для неї значення fMain.
2. Задати для форми такі властивості:
 - Text = "Лабораторна робота №7" (текстові значення вводять без лапок)
 - Name = fMain
 - StartPosition = CenterScreen
 - Icon->MaximizeBox = false

Щоб задати властивість будь-якого компонента, слід спочатку вибрати його у дизайнері форм, клікнувши на ньому один раз. Далі у вікні Properties слід знайти потрібну властивість (за її назвою) і ввести для неї значення. Вікно Properties завжди відображає перелік властивостей для вибраного в дизайнері форм компонента. Щоб вибрати форму, слід клікнути на ній у тому місці, де немає інших компонентів.

3. Перейти до вікна Toolbox, що розташоване зліва у середовищі Microsoft Visual Studio (рис. 7.3). Це вікно містить перелік компонентів, які можна використовувати при проектуванні застосунку. Зокрема, тут є багато компонентів, які представляють різні елементи керування для вікон програми (кнопки, списки, меню тощо). Кожен з цих компонентів описаний окремим класом (ці класи поставляються разом з Microsoft Visual Studio). Відповідно, кожен клас має властивий йому набір властивостей, методів та подій (з їх описом можна ознайомитися у довідці MSDN).

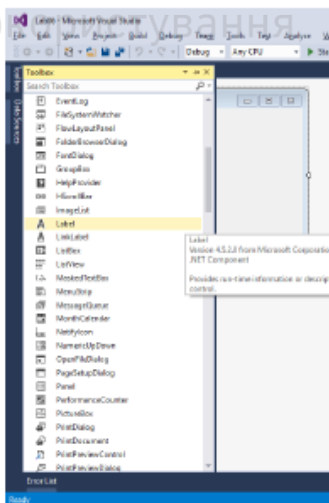


Рис. 7.3. Вікно Toolbox

¹ В загальному випадку назви змінним, які представляють об'єкти, можна давати довільно. Але прийнято, щоб назва містила інформацію про тип об'єкта та його призначення. У назві fMain префікс f означає тип об'єкта (це екземпляр класу Form), а Main означає, що це головна форма програми.

4. Розмістити на формі (у дизайнер форм) три компоненти Label (вони представляють текстові написи – "Змінна X1" та інші). Для цього знайти у вікні Toolbox цей компонент, клікнути на його кнопці, а потім клікнути у потрібному місці форми. На формі з'явиться вибраний компонент. В подальшому його можна перетягувати мишкою.

5. Для кожного елемента Label слід задати властивість Text, відповідно:

- "Змінна X1"
- "Змінна X2"
- "Результат розрахунку Y"

Для цього спочатку слід вибрати перший компонент, задати його властивість (у вікні Properties), потім вибрати другий і т.д.

6. Напроти трьох міток Label розмістити на формі три компоненти Textbox (вони представляють білі прямокутні поля для текстового вводу). Ці компоненти призначені відповідно для введення значень x_1 та x_2 та виведення значення функції. Попередньо розміщені компоненти Label пояснюють призначення цих полів.

7. Для кожного елемента Textbox слід задати властивість Name (назву), відповідно:

- tbX1²
- tbX2
- tbY

8. Для останнього елемента Textbox слід задати властивість ReadOnly = true (тільки для читання), оскільки його значення користувач не повинен змінювати безпосередньо.

9. У нижній частині форми розмістити три компоненти Button. Їх призначення – ініціювання таких дій програми: розрахунок, очищення полів та вихід.

10. Для кожної кнопки слід задати властивості Name (назва) і Text (напис), відповідно:

- для першої: btnCalculate³ та "Обчислити"
- для другої: btnClear та "Очистити"
- для третьої: btnExit та "Вихід"

11. У дизайнері форм задати розмір вікна таким, щоб воно охоплювало всі розміщені елементи керування та не містило зайвого порожнього місця з країв. Для цього слід перетягнути правий чи нижній край вікна, або правий нижній кут вікна. Щоб забрати відступ зліва або зверху, потрібно виділити всі компоненти (Ctrl+A у дизайнері форм) та перетягти їх вліво чи вгору.

12. Виконати пробний запуск застосунку, натиснувши клавішу F5. Після запуску відобразиться спроектоване вікно (рис. 7.4). У поля вікна, представлені елементом керування Textbox, можна вводити якийсь текст. На кнопки можна натискати, але при цьому не виконуються ніякі дії (бо нічого ще не запрограмовано). Вікно слід закрити, натиснувши на кнопку з хрестиком у правому верхньому кутку).

На цьому розробка вікна завершена.

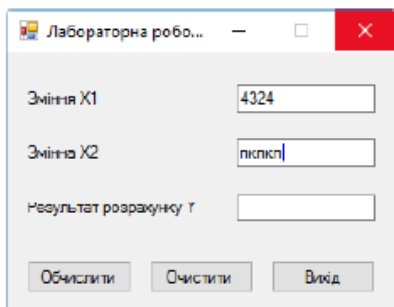


Рис. 7.4. Спроектоване вікно

² У назві tbX1 префікс tb означає тип компонента – Textbox, а X1 – призначення: ввід значення X1. Назви інших полів сформовано за цим же принципом.

³ У назві btnCalc префікс btn означає тип Button, а Calculate – призначення кнопки (calculate – обчислювати). Назви інших кнопок сформовано аналогічно.

Для ініціювання обчислення виразу призначена кнопка "Обчислити" (btnCalculation). Щоб запрограмувати дії, які виконуватиме програма при натисканні на кнопку, слід створити метод – оброблювач події Click:

1. У дизайнері форм виділити кнопку "Обчислити".
2. У вікні Properties натиснути на кнопку Events і перейти на сторінку відображення подій.
3. Знайти подію Click і два рази клікнути у полі справа від неї. У середовищі Microsoft Visual Studio відкриється вікно редактора коду з автоматично створеним методом – оброблювачем цієї події:

```
private void btnCalculation_Click(object sender, EventArgs e)
{
}
```

Щойно створений метод – оброблювач події не містить жодної інструкції. Код, написаний у цьому методі, автоматично виконуватиметься при кожному натисканні на кнопку "Обчислити" (тобто, при кожному настанні події Click).

4. Записати у метод btnCalculation_Click код з лістингу 7.1.

Лістинг 7.1

```
if (string.IsNullOrEmpty(tbX1.Text) ||
    (String.IsNullOrEmpty(tbX2.Text)))
{
    tbY.Text = "Не введено даних!";
    return;
}

double x1 = double.Parse(tbX1.Text);
double x2 = double.Parse(tbX2.Text);

double y = x1 * x2;

tbY.Text = y.ToString("0.#####");
```

Програмування дії очищення полів

Якщо у середовищі Microsoft Visual Studio відкрито редактор коду, то для переходу в дизайнер форм слід вибрати команду View Designer контекстного меню (чи натиснути клавіші Shift+F7), або скористатися відповідною вкладкою середовища. І навпаки: якщо відкрито дизайнер форм, то для переходу в редактор коду слід вибрати команду View Code контекстного меню (чи натиснути клавішу F7), або скористатися відповідною вкладкою середовища.

1. У дизайнері форм виділити кнопку "Очистити".
2. У вікні Properties натиснути на кнопку Events і перейти на сторінку відображення подій.
3. Знайти подію Click і два рази клікнути у полі справа від неї. У середовищі Microsoft Visual Studio відкриється вікно редактора коду з автоматично створеним методом – оброблювачем цієї події:

```
private void btnClear_Click(object sender, EventArgs e)
{
}
```

4. Записати у метод btnClear_Click код з лістингу 7.2.

	Національний університет водного господарства та природокористування	Лістинг 7.2
	<pre>tbX1.Text = string.Empty; tbX2.Text = string.Empty; tbY.Text = string.Empty;</pre>	

Програмування дії завершення роботи застосунку

1. У дизайнері форм виділити кнопку "Вихід".
2. У вікні Properties натиснути на кнопку Events і перейти на сторінку відображення подій.
3. Знайти подію Click і два рази клікнути у полі справа від неї. У середовищі Microsoft Visual Studio відкриється вікно редактора коду з автоматично створеним методом – оброблювачем цієї події:

```
private void btnExit_Click(object sender, EventArgs e)
{
}
}
```

4. Записати у метод btnExit_Click код з лістингу 7.3.

<pre>Application.Exit();</pre>	Лістинг 7.3
--------------------------------	-------------

Запуск і тестування роботи застосунку

Щоб запустити застосунок, у середовищі Microsoft Visual Studio потрібно натиснути клавішу F5. Далі слід ввести значення для змінних X1, X2, і перевірити функціонал всіх кнопок: для розрахунку, очищення форми та виходу з програми.

Завдання для самостійного опрацювання

1. Вирішити розглянуте раніше завдання для заданої функції $f(x_1, x_2)$ згідно варіанту (таблиця 7.1).
2. Додатково до попереднього завдання, обчислити та вивести на екран:
 - для варіантів, номери яких закінчуються на 0...3 – середнє арифметичне значень x_1, x_2 ;
 - для варіантів, номери яких закінчуються на 4...6 – менше із значень x_1, x_2 ;
 - для варіантів, номери яких закінчуються на 7...9 – більше із значень x_1, x_2 ;
3. При виводі числових значень на екран:
 - для варіантів, номери яких закінчуються на 0...3 – виводити 4 знаки після коми;
 - для варіантів, номери яких закінчуються на 4...6 – виводити 3 знаки після коми;
 - для інших варіантів – виводити значення в експоненційному форматі.



Варіант	Завдання	Варіант	Завдання
1.	$y = \frac{\sqrt{x_1^3 + x_2^5}}{1000 \cdot \sqrt{x_1 + x_2^5}} + 65$	2.	$y = 23 \cos^2(x_1^3 x_2^5) + 2x_1$
3.	$y = \frac{\sqrt{\cos^3(x_1) + x_2}}{x_1^{13} + \frac{3}{\cos(x_2)}}$	4.	$y = \sin(x_1 - x_2^3 + \sqrt{x_1}) - 1.3x_1^3$
5.	$y = \frac{4 \sin(3 + x_1 x_2)}{34 - 9x_2^3}$	6.	$y = \sqrt{56x_1 + \frac{x_1 + x_2 + \sin(x_1 x_2)}{5 - \cos(x_2^2)}}$
7.	$y = \sqrt[4]{0.1x_1 \sin x_2 \cos x_1^2 + 55x_1 x_2}$	8.	$y = \sqrt{\frac{\cos(2x_2) + x_1 / x_2}{16x_2 x_1}}$
9.	$y = \cos(\sqrt{x_2} + 34x_1) - 4 \sin(x_2)$	10.	$y = \sin^2\left(x_1 \frac{x_2}{x_1 + 53x_2^2}\right)$
11.	$y = \frac{\sqrt{x_1^3 + x_2^5}}{1000} + 65$	12.	$y = \exp(x_1 - x_2^2) + 31.55x_2 x_1^2$
13.	$y = \frac{6 - \cos(3 + x_1)}{34 - 9x_2^3 + x_2}$	14.	$y = \cos^4\left(x_1 - \sqrt{\frac{x_2}{x_1 + 53x_2^2}}\right)$
15.	$y = 23 \sin^2(x_1^3 x_2^5) + 2x_1 + \cos(x_1 x_2)$	16.	$y = \sqrt{56 + \frac{x_1 + x_2 + \sin(x_1 x_2)}{5 \cos(x_2^2)}}$
17.	$y = 45x_1 \sin x_2 + \sqrt{9x_2 x_1^3}$	18.	$y = \frac{1}{4 + x_2} \cdot \sqrt{\cos^2\left(\frac{x_2}{x_1}\right)}$
19.	$y = \frac{\cos^3(x_1) + 45 + x_2}{x_1^{13} + \cos(x_2)}$	20.	$y = \frac{5\sqrt{x_1^3 + x_2^5} - \cos(x_2)}{\exp(x_1)}$
21.	$y = \frac{\cos^2\left(\lg_{10} \frac{x_2}{x_1}\right)}{45 + x_2}$	22.	$y = 45 \sin(x_1 + x_2 + \lg_{10}(x_1 x_2^2))$
23.	$y = \sqrt{\frac{x_2^2 + x_1 / x_2}{16x_2 x_1}}$	24.	$y = \frac{\ln(x_2)}{\sqrt[4]{0.6x_1 \sin x_2 \cos x_1^4}}$
25.	$y = \cos^3\left(\exp\left(\frac{x_1 + 2x_2 + 9}{0.666}\right)\right)$	26.	$y = \frac{3x_2 - x_1^2}{\cos^3\left(\frac{x_1 + 2x_2 + 9}{0.37}\right)}$
27.	$y = \cos(\sqrt{x_2} + 34 \cdot \sin(x_1)) - 4 \sin(x_2)$	28.	$y = \sqrt{\frac{x_2^2 + x_1 / x_2}{\cos(x_1^3 x_2^5) + 2x_1}}$
29.	$y = 0.1x_1 \sin x_2 \cos x_1^4 + 55$	30.	$y = \frac{\sin^3(x_1) + 45 + x_2}{2x_1^4 + 4x_2}$